231839

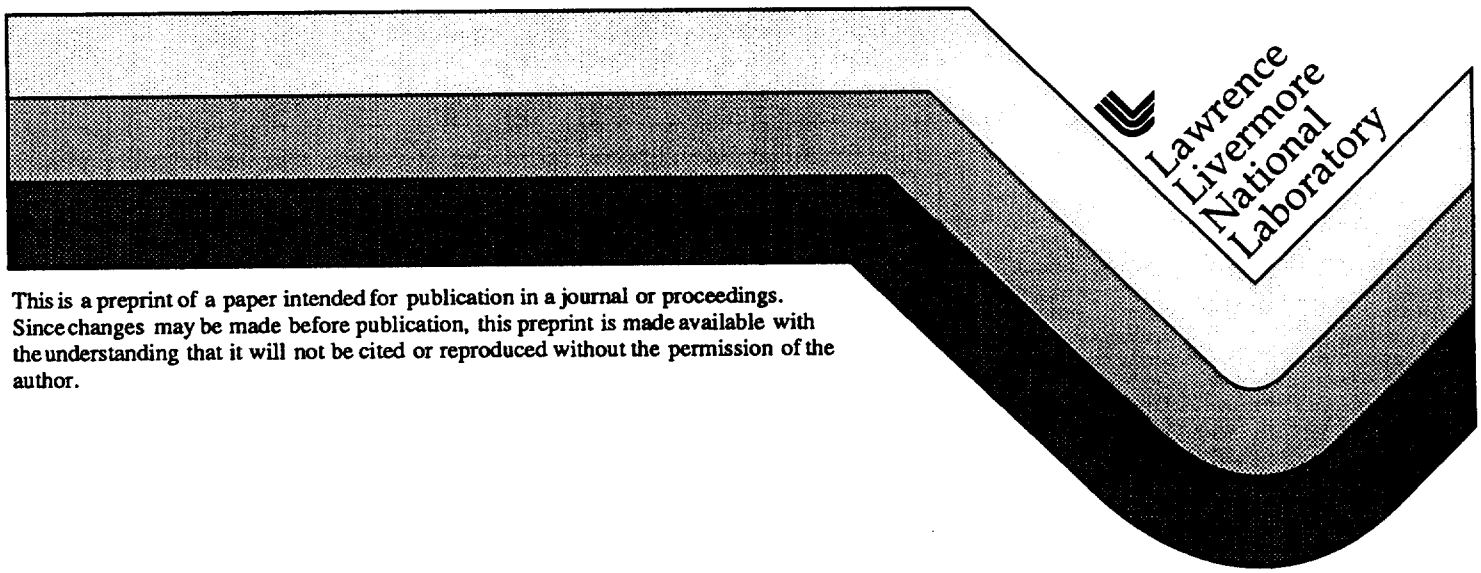
UCRL-JC-125948 Rev. 1
PREPRINT

Coupled Ocean/Atmosphere Modeling on High-Performance Computing Systems

P.G. Eltgroth
J.H. Bolstad
P.B. Duffy
A.A. Mirin
H. Wang
M.F. Wehner

This paper was prepared for submittal to the
Parallel Processing for Scientific Computing Conference
Minneapolis, MN
March 14-17, 1997

March 1997



Coupled Ocean/Atmosphere Modeling on High-Performance Computing Systems*

P.G. Eltgroth† J.H. Bolstad† P.B. Duffy† A.A. Mirin† H. Wang†
M.F. Wehner†

Abstract

We investigate performance of a coupled ocean/atmosphere general circulation model on high-performance computing systems. Our programming paradigm has been domain decomposition with message-passing for distributed memory. With the emergence of SMP clusters we are investigating how to best support shared memory as well. We consider how to assign processors to the major model components so as to obtain optimal load balance. We examine throughput on contemporary parallel architectures, such as the Cray-T3D/T3E and the IBM-SP family.

1 Introduction

In order to model the Earth's climate system at a satisfactory level of detail, one needs substantially more computing power than is generally available on one, or even a few, modern computer processing units. Thus we have developed a climate computer code which is designed to run on Massively Parallel Processors (MPPs). The code is portable so that, as new computer architectures become available, it can quickly be transferred to them. The code is written primarily in Fortran, but substantial pieces of the support system are written in C. The portability aspects of dynamic memory management and communications are handled through a macro interface which encapsulates architecture specific details [2]. The programming style is that of message passing. Explicit control is exercised over the transfer of data across processors. In the case where shared memory is available and efficient, the macros have been modified to use it.

The parallel strategy exploits both functional and domain decomposition. The functional decomposition allows an arbitrary number of processors to be applied to each of several different physics modules or packages, such as the ocean or atmosphere. When appropriate, one processor is dedicated to framework (non-physics) tasks. Some of the modules which have been implemented underneath the framework are atmosphere dynamics [1], [12] - [15], ocean dynamics [6], [15], ocean sea ice [5], ocean biogeochemistry [3], land surface processes [9], and atmospheric chemistry [7], [8]. Coupling can be sustained at the surface between subsystems or throughout the entire volume for interpenetrating physical systems. This report treats the atmosphere-ocean-ice coupled system. Coupling is effected by explicit exchange of information across the ocean-atmosphere interface. The resolutions of the atmosphere and ocean may be different in horizontal directions. At present, the sea ice module uses the same horizontal spatial resolution and time step as does the ocean.

*This is LLNL Report Number UCRL-JC-125948. Work performed under U.S.D.O.E. Contract W-7405-ENG-48.

Domain decomposition is two dimensional, with variables distributed among processors in latitude-longitude blocks. All vertical levels in a given latitude-longitude block are handled by the same processor.

2 Parallelism

On the fastest computers generally available, it can take months to complete long climate change experiments. MPP technology has enabled stand-alone atmospheric and oceanic Global Circulation Models (GCMs) to calculate heretofore unresolvable phenomena and has significantly advanced our science understanding, e.g. [10]. Coupled ocean-atmosphere-land models will benefit from similar increases in computational performance. The ability to run very large problems can be just as important as increased speed.

Our coupled climate model essentially joins existing parallel GCMs for atmosphere and ocean. The number of processors assigned to each module is fixed at run time, as is the number of domains. A single processor is assigned to tasks related to framework issues (e.g., some machines have restrictions on input/output.) As implemented, there is a one-to-one correspondence between processors and domains in each module. That is, no processor updates more than one domain in one physics module. For the ocean package, domains which lie entirely over land are not mapped to a processor. This avoids wasting an unused resource.

The number of horizontal cells in each domain is made to be as uniform as possible. For the atmosphere, this gives a total number of cells in each domain which is close to a mean value. For the ocean, because of bottom topography and the possibility of land cells, the number of active cells per processor can vary significantly. For both ocean and atmosphere, since cells are based on latitude-longitude coordinates, the physical size of the cells becomes smaller near the poles. In order to avoid small time steps in the physics advance, the GCMs are run with the same (large) time step everywhere. Then unstable modes of behavior are removed near the poles by a filtering process. The filtering involves communication over lines of constant latitude. Since this filtering would unbalance each GCM, the processors for unfiltered domains are used to help the "overworked" processors near the poles. This remapping of work involves communication across different latitudes, and is a tradeoff of increased communication for better load balance.

Each individual code execution uses a static arrangement of processors and domains. A single restart file, independent of domain decomposition, is written for each GCM. As a given problem progresses, the arrangement of processors and domains can be changed at restart time by simply changing an input file.

3 Coupling Strategy

The atmosphere-ocean coupling is designed around an explicit exchange of information between the two GCMs. A partial list of information provided by the atmosphere includes wind stress, low level air temperature, sea level pressure, humidity, precipitation, and radiant energy fluxes. A partial list of ocean information includes ocean current, sea surface temperature, ice/snow parameters, surface roughness, and albedo. In the near future, some of these parameters may be found in a consistent manner using primitive data from each of the GCMs. At that time, it might be most efficient to assign the coupling section of the code to its own set of processors (and, perhaps, domains.) However, for the work reported here, all of the coupling preparation and utilization is done on the processors assigned to the individual GCMs.

Our approach to the design of a parallel coupled climate model [4] can be contrasted to that of others [11]. We have attempted to minimize the overhead associated with coupling. In our model, data is communicated between modules by message passing within the same executable code object. Direct message passing, rather than transfer through intermediary files, offers significant performance advantages. Furthermore, through a sorting algorithm determined at problem initialization, message traffic is routed directly between geographically overlaying subdomains of the atmosphere and ocean models.

The coupling data is allowed to have any resolution appropriate to the originating physics package. Data is interpolated or area weighted to the new grid as part of the exchange process. Typically, the ocean model has finer grid resolution than the atmosphere model. The exchange of information is carried out as a logically separate process from the update of the GCM state and is scheduled before the update. In the present version of the code, each module reports instantaneous values of variables to its exchange partners. The schedule for exchange may be set independently from the time advance phase. For results presented here, the exchange of information is symmetric; i.e., the atmosphere sends to the ocean at the same simulated time as the ocean sends to the atmosphere. The code framework allows an arbitrary and non-symmetric exchange. Future work will investigate a more general treatment of exchanged information.

4 Performance and Load Balance on the T3D

The coupled atmosphere-ocean-ice climate code has been run extensively on Cray T3D computers. Results from other machines will not be included in this publication, but they may be reported in the conference presentation. The largest number of processors considered in this report is 128. This restriction is primarily due to local scheduling policies.

A variety of resolutions have been examined in coupled mode. The coarsest horizontal resolution uses 5° in longitude by 4° in latitude for both atmosphere and ocean. The atmosphere vertical resolution is 9 levels and that of the ocean is 20 levels. The finest resolution tested was 5° in longitude by 4° in latitude and 9 levels for the atmosphere together with 2.5° in longitude by 2° in latitude and 24 levels for the ocean. Runs were also performed with a 5° in longitude by 4° in latitude atmosphere and a 3° in longitude by 3° in latitude ocean. This last configuration tested the code for non-commensurate resolutions. Note that a change of a factor of 2 in linear horizontal resolution means roughly a change of a factor of 8 in the number of operations performed because the time step generally must be changed by the same factor in order to maintain a stable calculation.

All runs reported here were carried out for substantial lengths of simulated time. Thus problem initiation and initial cache performance have little effect on timing results. Figure 1 shows individual GCM performances for the T3D. In this example, the resolution of the ocean model is 3° by 3° by 15 levels, and the resolution of the atmospheric model is 5° by 4° by 9 levels. The domains used in each model were arranged to be roughly square. We use the notation A6x7 to denote a grid of 6 domains along the longitude direction by 7 domains along the latitude direction for the atmosphere. The letter O is used to denote ocean. The number of processors used is the number of domains plus one minus the number of domains entirely over land (for the ocean only.)

The measure of performance shown is the amount of model time simulated per machine second. In a perfectly parallelizable model, the curves would be linear. In practice, there is degradation due to communications and load imbalance. By inspection of Figure 1, it can be seen that, in the absence of any interaction between atmosphere and ocean, for a

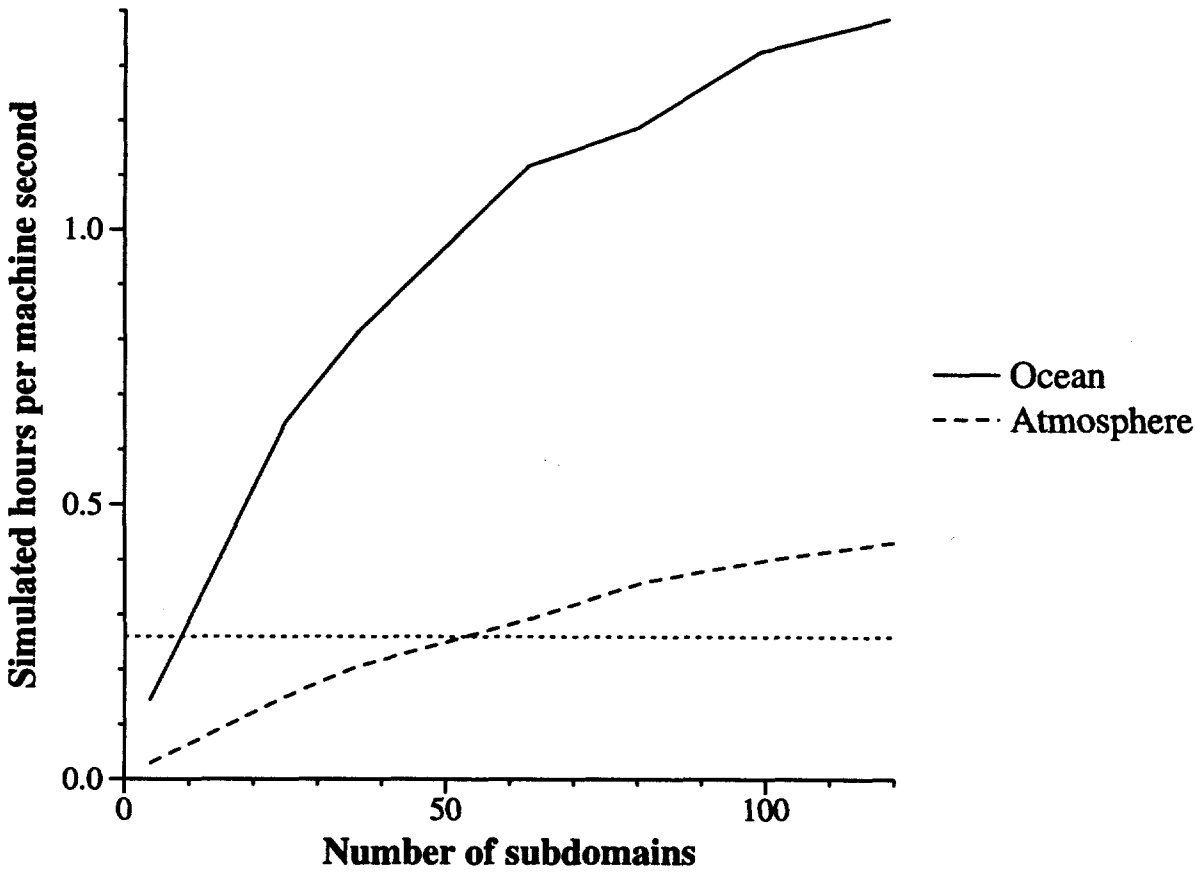


FIG. 1. *Performance of separate atmosphere and ocean General Circulation Models as a function of the number of subdomains, measured on the Cray T3D. The horizontal dotted line indicates a "balanced" situation in which the two models complete a simulation in the same time.*

total of 64 processors, the two calculations will be balanced (consuming the same wallclock time) at a ratio of approximately 6 atmosphere processors to 1 ocean processor. As the number of processors grows, the ratio becomes even larger.

As a result of the simple types of considerations given above, the coupled code was designed to place the majority of the coupling calculations (such as variable preparation and remapping) on the ocean processors. This should move the balance point to a smaller ratio. The additional cost of communication is, of course, shared between modules and will slow the entire calculation, possibly affecting the balance point in a way that depends on machine architecture. The Figure 1 results were prepared with roughly square domain decompositions, but these are usually not optimal for the individual GCMs. The reason is that, while square subdomains minimize the nearest neighbor type of communication, other types of operations (such as filters) involve communications predominantly along lines of constant latitude. This tends to favor domains which cover a large span of longitude. A very important effect arises for domain configurations which place different numbers of cells in

the domains. We do not demand that the total number of cells be exactly evenly distributed among processors. The algorithm for distribution attempts to minimize the effect, but, for a small number of cells and a large number of domains, the resulting granularity can cause non-negligible load balance problems.

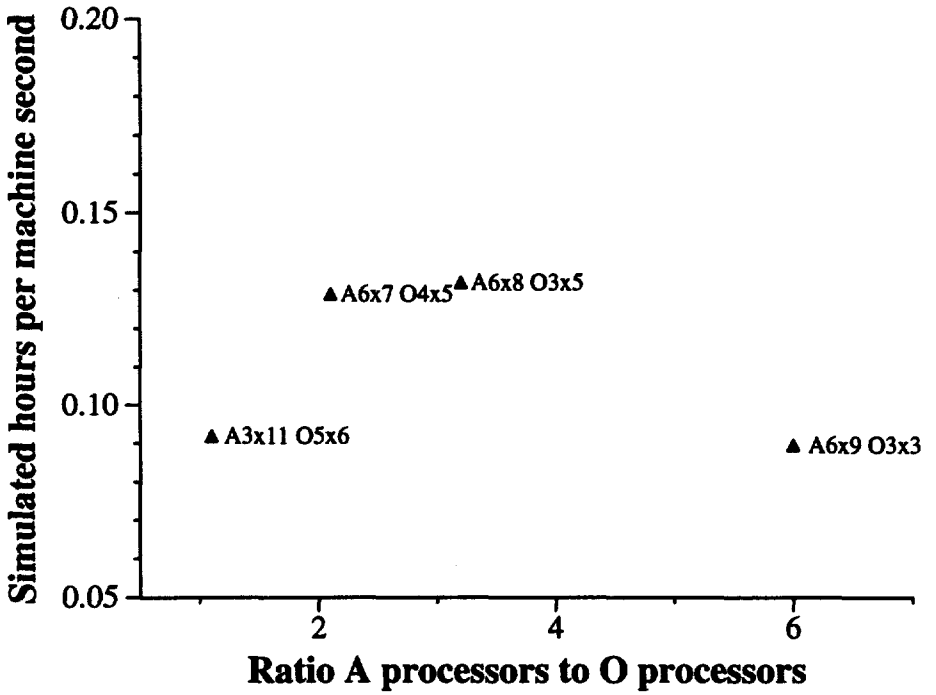


FIG. 2. *Performance of coupled atmosphere and ocean General Circulation Models as a function of the ratio of the number of atmosphere processors to the number of ocean processors, measured on the Cray T3D. The total number of processors was either 63 or 64, and the domain configurations are shown next to the points.*

Figure 2 shows the relative performance of various coupled code runs versus the ratio of atmosphere processors to ocean processors. All runs have a total of 62 or 63 processors, thus using essentially all of a 64 processor partition on the Cray T3D. The physical resolution was the same as that used to obtain Figure 1. The best performance occurs near a ratio of 3 to 1 for atmosphere processors to ocean processors. The absolute performance for the coupled calculation has fallen by approximately 50% relative to the idealized uncoupled calculation. This appears to be due to the extra arithmetic involved in the coupling process as well as to communication imbalance. The performance picture is also muddled by the presence of additional output to track coupling variables, as well as by changes in systems, compilers, libraries, and the like. In order to have a realistic estimate of the cost of a long run, the test runs were carried out for a substantial simulated time, so that input/output burden would be accurately represented.

One coupled run was completed using a total of 127 processors (A6x16 O5x6). It showed a performance of 0.187 simulated hours completed per machine second. This is substantially better than the 64 processor performance and agrees with the scaling suggested by the uncoupled results. In order to complete a 30 year simulation at the above rate, about 400 hours of computer time would be required. The next generation of MPPs may bring this

number down to a less heroic value.

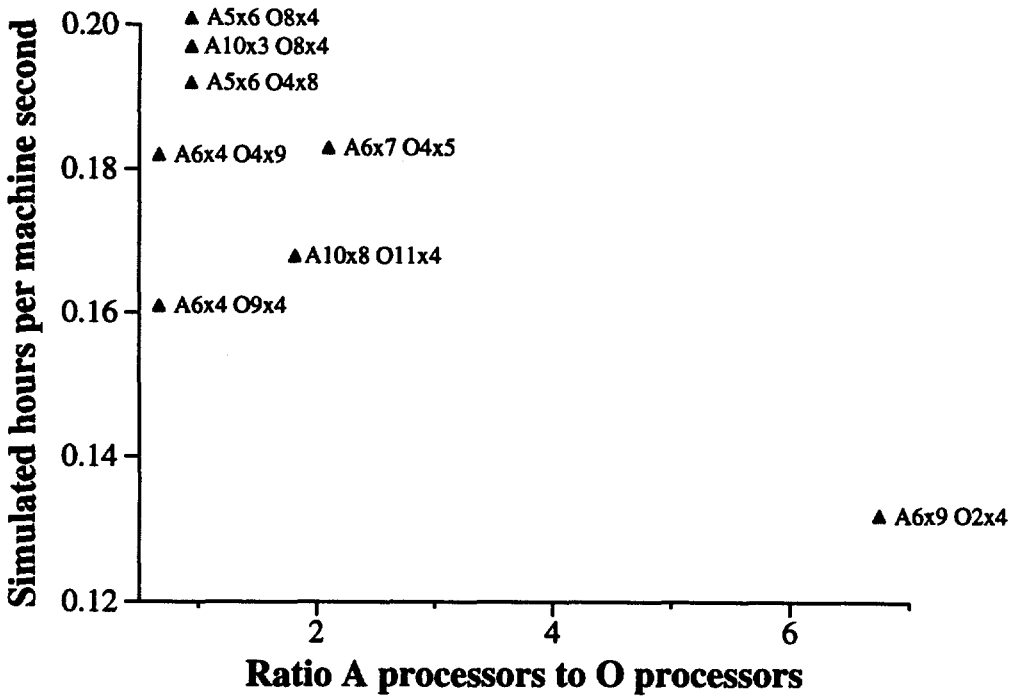


FIG. 3. *Performance of coupled atmosphere and ocean General Circulation Models as a function of the ratio of the number of atmosphere processors to the number of ocean processors, measured on the Cray T3D. For these runs, the horizontal resolutions were the same (see text.) The total number of processors varies, and the domain configurations are shown next to the points.*

The last set of calculations reported here was carried out on an "equal resolution" configuration, with horizontal resolution of both atmosphere and ocean fixed at 5° in longitude by 4° in latitude. The atmosphere had 9 levels and the ocean 20. The match of horizontal resolution was intended to simplify the exchange of information across modules, thus giving the least possible burden to the coupled calculation. Results are shown in Figure 3. Note that the total number of processors is not fixed for Figure 3. In particular, the performance for configuration A10x8 O11x4 is substantially below values for smaller numbers of processors. This indicates that the coarse resolution calculation does not have enough cells to use a large number of processors efficiently. However, the ocean has a larger number of cells in the vertical which partially offsets the effect of fewer cells in the horizontal.

The best performance for the coarse horizontal resolution case appears to occur for a roughly equal number of ocean and atmosphere processors. The observed performance indicates that this problem would complete a 30 year simulation in about 250 hours of T3D time. It is interesting to note that two experiments varying the latitude-longitude domain decomposition for the ocean module do not give a clear signal that one type of decomposition is always best. It is certain that more experimentation will be needed to determine the reasons for the observed behavior. Normal performance tools appear to be inadequate for a code this complex and large.

5 Conclusions

We have begun to map out the issues which influence the performance of large coupled codes on MPPs. At present, we have measured only a relatively coarse set of parameters. It is clear that more detailed, but not disruptive, information will have to be acquired before we have the ability to predict an optimal balance of processors for a given work load. Particular attention must be paid to the issue of accounting for time spent in the preparation and transmission of information between physics modules. We have observed that, for coarsely resolved problems, this overhead can be a significant cost.

References

[1] A. Arakawa and V. Lamb, *Computational Design of the Basic Dynamical Processes of the UCLA General Circulation Model*, Methods in Computational Physics, 17 (1977), pp. 173–265.

[2] J. Brown and A. A. Mirin, *MICA, a Facility to Achieve Portability for Message Passing and Dynamic Memory Management in FORTRAN*, NERSC Buffer, 18 (1994), pp. 2–12.

[3] P. B. Duffy, J. Amthor, K. Caldeira, P. S. Connell, D. E. Kinnison, J. Southon, and D. J. Wuebbles, *The Global Budget of Bomb 14C*, submitted to Climatic Change (1997).

[4] A. Mirin et al., *Climate System Modeling using a Domain and Task Decomposition Message-Passing Approach*, Computer Physics Communications 84 (1994), pp. 278–296.

[5] J. M. Oberhuber, Journal of Physical Oceanography 23 (1993), p. 808.

[6] R. Pacanowski, K. Dixon, and A. Rosati, *The GFDL Modular Ocean Model Users Guide version 1.0*, GFDL Ocean Group Technical Report #2 (1991).

[7] D. A. Rotman, *Parallel Computing in Atmospheric Chemistry Models*, UCRL-JC-123575, Lawrence Livermore National Laboratory (1995).

[8] D. A. Rotman, D. J. Wuebbles, and J. E. Penner, *Atmospheric Chemistry Using Massively Parallel Computers*, AMS Fifth Annual Symposium on Global Change Studies (1994)

[9] P. Sellers et al., *A revised Land-Surface Parameterization (SiB2) for Atmospheric GCMs. Part 1: Model formulation.*, Journal of Climate 9 (1996), pp. 676-705.

[10] A. Semtner, *Modeling Ocean Circulation*, Science 269 (1995), pp. 1379–1384.

[11] E. Sevault, P. Noyret, L. Terray, and O. Thual, *Distributed and Coupled Ocean-Atmosphere Modeling*, Proceedings of the 6th Workshop on Use of Parallel Processors in Meteorology, ECMWF, Reading, England (1994), p. 512

[12] M. F. Wehner, A. A. Mirin, P. G. Eltgroth, and W. Dannevik, *Toward a high performance distributed memory climate model*, in Proceedings of the 2nd International Symposium on High Performance, Distributed Computing (1993), p. 102.

[13] M. Wehner and C. Covey, *Description and validation of the LLNL/UCLA parallel atmospheric GCM*, UCRL-ID-123223, Lawrence Livermore National Laboratory (1995).

[14] M. F. Wehner, A. A. Mirin, P. G. Eltgroth, W. Dannevik, C. R. Mechoso, J. Ferrara, and J. Spahr, *Performance of a Distributed Memory Finite Difference Atmospheric General Circulation Model*, Parallel Computing 21 (1995), pp. 1655–1675.

[15] M. F. Wehner, A. A. Mirin, P. G. Eltgroth, and W. Dannevik, *Climate Systems Modeling on Massively Parallel Computers at Lawrence Livermore National Laboratory*, Proceedings of the NGEMCOM Workshop, National Environmental Supercomputing Center, Bay City MI. (1995).